Defect Probability of Directed Self-Assembly Lithography: Fast Identification and Post-Placement Optimization

Seongbo Shim, Woohyun Chung, and Youngsoo Shin
Department of Electrical Engineering, KAIST
Daejeon 305-701, Korea

***Abstract*—In directed self-assembly lithography (DSAL), an inter-cell cluster of contacts, which crosses the boundary of cells, is more likely to cause patterning failure because corresponding guide pattern (GP) has not been verified beforehand. All forms of inter-cell clusters can systematically be identified and grouped, which allows us to define DSA defect probability when two arbitrary cells are placed side by side. We then address post-placement optimization, in which some cells are flipped and some cells are swapped with their adjacent cells so that the number of whitespaces inserted in between cell pairs of high defect probability is minimized. Experiments with a few test circuits demonstrate 11% increase of placement density, on average, with no expected DSA defects.**

I. Introduction

In sub 10-nm technology, contacts are smaller than the resolution limit of traditional optical lithography. A directed self-assembly lithography (DSAL) has been proposed as an alternative patterning solution [1], [2]. In DSAL, contacts that are physically close are patterned together in two steps. (1) A contour that surrounds the contact cluster, called a guide pattern (GP), is synthesized on a mask [3], [4], which goes through optical lithography to form a GP image on a wafer (see Fig. 1). (2) GP is filled with block copolymers (BCPs), which are strings of two types of polymer. BCP in its nature is arranged as shown in Fig. 1(c) due to forces between polymers and GP contour. One type of polymer is etched away, which leaves final contacts.

A contact that is not correctly patterned in these steps is called a DSA defect. Some defect examples are shown in Fig. 2: too large contact that may cause electrical short, target contact is not open due to too small contact image on a wafer, and unexpected contact appears during DSA process. Since contacts are patterned through two independent steps, DSAL involves more defects than optical lithography does. Our preliminary experiments show that about 1∼2% error in GP causes 4∼5% error in the final contact.

The probability that DSA defect occurs, called DSA defect probability (or simply defect probability), usually increases as a cluster contains more contacts because corresponding GP contour becomes more complex. Such clusters may be split into smaller ones by relocating some contacts while standard cells are designed [5]. However, when two cells are located side by side (e.g. during placement), large clusters can still form across the cells as illustrated in Fig. 3. Verifying the correctness of all inter-cell clusters[1] is very important but difficult. Applying lithography- and DSA-simulations on whole layout may be considered as a possibility, but impractical amount of time does not allow this approach.

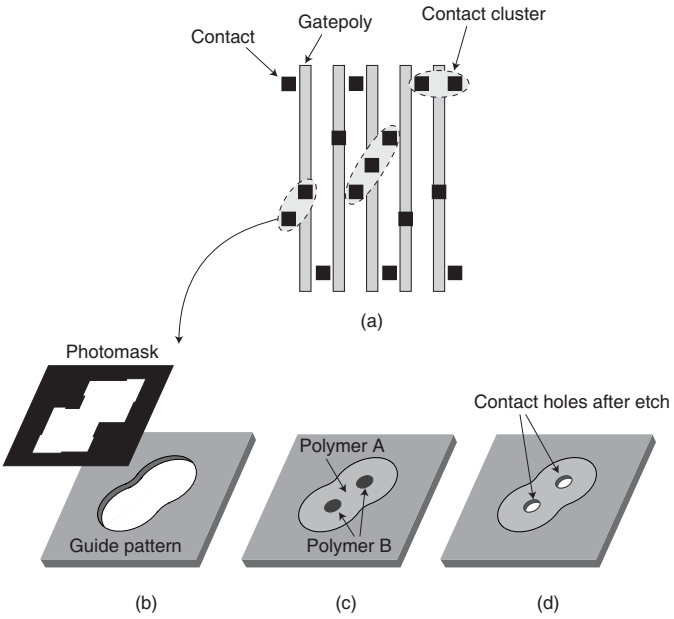

Fig. 1. DSAL process: (a) contacts are clustered, (b) GP is formed on a wafer (through optical lithography), (c) GP is filled with BCPs, which are self arranged, and (d) contacts are formed after one type of polymer is etched away.

Alternative would be to guarantee the correctness of inter-cell clusters in design time; this is the motivation of our paper. We consider two components for this approach:

- Determine defect probabilities of all cell pairs in advance, when two cells of each pair are placed side by side. (Section II)
- After standard placement, we perform post-placement

[1]Inter-cell clusters do not form between the cells that are adjacent above and below due to the presence of power rails.

978-1-4673-8388-2/15/$31.00 ©2015 IEEE

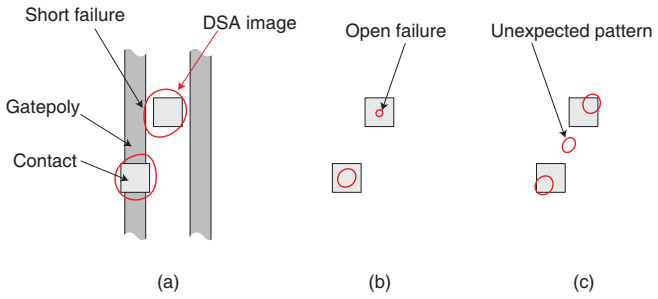

Fig. 2. Examples of DSA defect: (a) too large contact may cause electrical short, (b) actual contact is not open, and (c) unexpected contact appears.

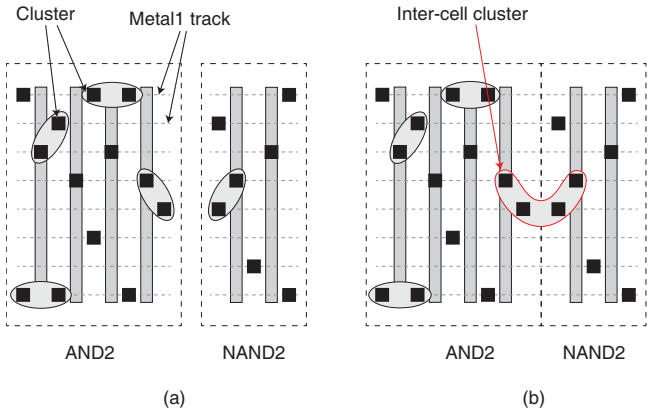

Fig. 3. Contact clusters (a) before- and (b) after-placement.

optimizations with the goal of minimizing the use of whitespace[2]. (Section III)

For the first component, we identify the inter-cell clusters of all cell pairs. Fortunately, there are many identical clusters, so the number of unique clusters is quite manageable. Each cluster is submitted to repeated lithography- and DSA-simulations; we utilize the simulation results to define the defect probability of a cluster, which we extend to define that of a cell pair. For post-placement optimizations, we consider two methods: (1) flip some cells (along y-axis) and (2) swap some adjacent cells as well as flip some. The second method benefits more but at the cost of increased wirelength due to larger cell displacement. We apply either method to each placement row, and corresponding problem can be formulated as the shortest path problem of directed acyclic graph with nonnegative weights, which can be solved in linear time. The application of the methods on a few test circuits indicates 11% increase of placement density.

The remainder of this paper is organized as follows. DSA defect probability is introduced in Section II; how to obtain its value for all cell pairs is addressed; its application to 10-nm synthetic library is demonstrated. In Section III, we presents

[2]Inter-cell cluster does not form if a whitespace of 1-poly pitch width is inserted.

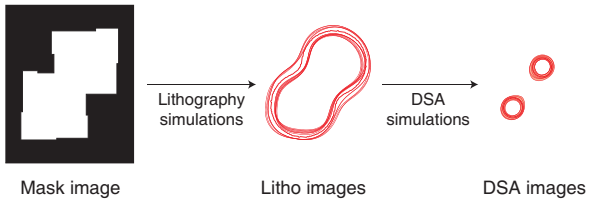

Fig. 4. Mask image of a GP, expected shapes of GP after optical lithography (lithography images) under various conditions, and expected shapes of contacts (DSA images) after DSA process.

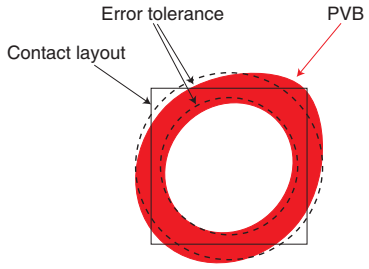

Fig. 5. DSA process variation band of a contact with error tolerance.

post-placement optimizations and its application to a few test circuits. The paper is summarized in Section IV with some candidate topics for further study.

II. DSA Defect

A. DSA Defect Probability

The probability that a particular contact has a defect after DSAL is obtained through repeated lithography- and DSA-simulations. As illustrated in Fig. 4, a lithography simulation with a mask image of a GP yields an expected GP shape on a wafer, called litho image. To account for lithography variations, the simulation is in fact repeated while lithography parameters (e.g. scanner focus, exposure energy, and mask manufacturing error) are varied [6], [7]; the result is a set of litho images. Each litho image is then submitted to a DSA simulator, which outputs an expected shape of a contact (or contacts), called DSA image. Each contact, in the end, is associated with multiple DSA images as shown in Fig. 4.

The region bounded by the outermost- and innermost-contours of multiple DSA images is called DSA process variation band (PVB) as illustrated in Fig. 5. The size of DSA image mainly determines the defect as shown in Fig. 2, and PVB indicates how much that size varies. So, defect probability can be defined by comparing the PVB area (red region) with some target region. We consider a circle inscribed in the contact layout; two concentric circles are then constructed with their radii corresponding to some error tolerance (e.g. $\pm 10\%$ in our experiment); the region between the concentric circles will serve as a target region. If whole PVB is contained within

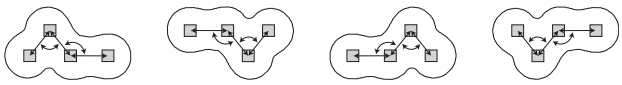

Fig. 6. An example of identical clusters of contacts.

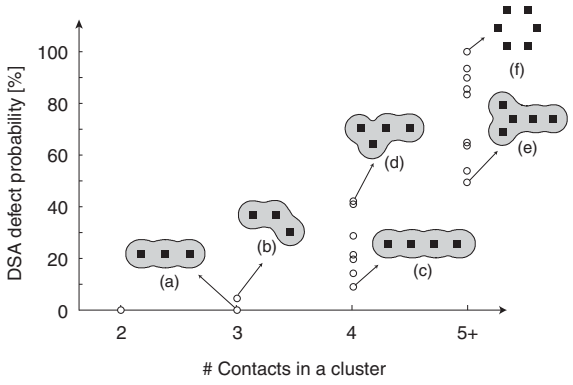

Fig. 7. DSA defect probability for various contact clusters.

the target region, defect is not expected. Otherwise, we define DSA defect probability by

$$D = \frac{\text{Area of PVB beyond tolerance}}{\text{PVB area}} \times 100\%. \qquad (1)$$

The defect probability of a contact cluster is assumed to be the maximum defect probability of its member contacts. The defect probability of a cell pair (see Fig. 3(b)) is represented by the maximum defect probability of inter-cell clusters.

DSAL consists of two steps, lithography and DSA process. The latter will also be affected by some variations sources. Unfortunately, there has been no study on those sources, and so DSA simulators with DSA process variations are currently unavailable. The PVB will inflate if those simulators become available and are used, and defect probability (1) will then increase.

B. Fast Identification of DSA Defect Probabilities

DSA simulation is a lengthy process, e.g. one simulation run on a contact cluster takes about 10 minutes. It is thus impractical to repeat the simulations for each cell pair one by one.

Fortunately, DSA process (Fig. 1(c) and (d)) is localized within a GP [8] implying that the clusters of same contact topology (the number of contacts, distances between adjacent contacts, and the shape of line connecting the contacts) are associated with the same GP. Therefore, we list all inter-cell clusters from all possible cell pairs in a library, and extract identical clusters by examining their contact topologies as shown in Fig. 6. Due to the Gridded Design Rule (GDR) [9], contact is allowed directly on a gate poly, or at the center of

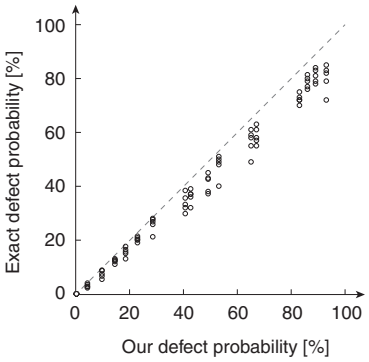

Fig. 8. Defect probability of exact method compared with that of our method.

two nearby gatepolys when it is placed along the metal 1 track (see Fig. 3). This limits diversity of contact topology, which makes many inter-cell clusters identical. So, the number of unique ones is small enough to handle.

Even though some contact clusters are topologically identical, the PVBs of corresponding GPs may be different due to optical interference with neighboring clusters. For each set of identical clusters, we pick two whose neighboring clusters are mostly dense and mostly sparse; we pick the cluster whose defect probability is larger as a representative.

C. Experiments

We demonstrate our method using a 10-nm synthetic library; layouts from 15-nm NanGate library [10] are shrunk so that the layouts can follow the GDR of 10-nm technology [11], in which the minimum contacted-poly pitch (CPP) is 45nm, size of contact is 22nm, and metal 1 track pitch is 36nm. ArF immersion lithography (1.35 NA) with annular illumination was assumed, and we use self-consistent field theory (SCFT) based DSA simulator [12], [13].

Our simple library contains 76 cells, so the total number of cell pair configuration is 23104 (4×76^2) given that each cell can be flipped. Contacts which are separated within a distance of 57nm are clustered, e.g. contact pair with horizontal distance of CPP. We identify about 40000 inter-cell clusters, but unique clusters are only 19. Defect probability calculation takes about 50 hours, which however is required only once.

Experiments indicate that the defect probability increases as a cluster contains more contacts as shown in Fig. 7. There is a substantial variation of defect probability even for the clusters of the same number of contacts. Cluster (a), (c), and (e) have lower defect probability. This is because of symmetry that GP has, which makes the cluster less sensitive to lithography variations because BCPs tend to be aligned periodically for lower energy. Due to the absence of symmetry, cluster (b) and (d) have high defect probability. Cluster (f) cannot be patterned by DSAL since corresponding GP is not synthesized, so its defect probability is 100%.

To assess the accuracy of our method, we verified 90 clusters; 5 clusters were randomly picked from each set of

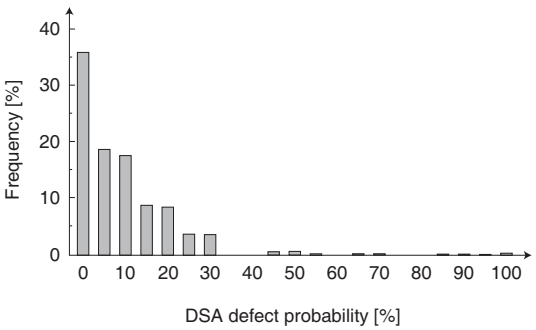

Fig. 9. A histogram of DSA defect probabilities of all cell pairs.

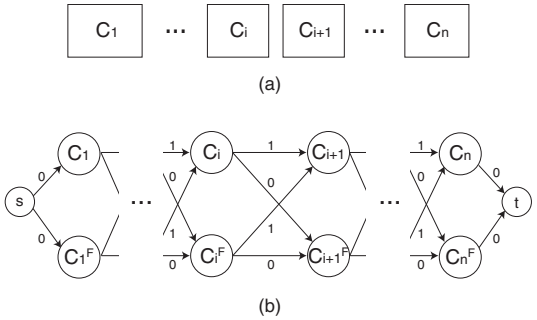

Fig. 10. (a) A placement row of n cells and (b) graph modeling to determine optimal cell flipping.

identical clusters (except cluster (f) in Fig. 7). Exact defect probability is obtained by repeated lithography- and DSA-simulations with 500 lithography settings, which are randomly sampled; the ratio of the number of DSA images contained within the tolerance region is considered as the value. Defect probabilities of our method are compared with the exact values as shown in Fig. 8, which indicates that our method predicts higher value, which is safe in manufacturing perspective.

Defect probabilities of all cell pairs are summarized as a histogram in Fig. 9. About 35% of pairs have zero defect probability: 17% pairs have no inter-cell clusters; 11% have clusters of two contacts, which have zero defect probability as shown in Fig. 7; the remaining 7% have clusters of three contacts that are linearly aligned (cluster (a) in Fig. 7), which also have 0% defect.

III. Post-Placement Optimization

Suppose that placement has been performed with some density target. A maximum DSA defect probability that is allowed is given as a threshold (usually this will be 0%). If there is a cell pair whose defect probability exceeds the threshold, a whitespace needs to be inserted in-between so that no inter-cell cluster can form. The goal of post-placement optimization is to minimally perturb the placement so that the amount of whitespace we insert is minimized. If whitespace is not enough even after the post-placement optimization, we repeat the placement and optimization with lower density target.

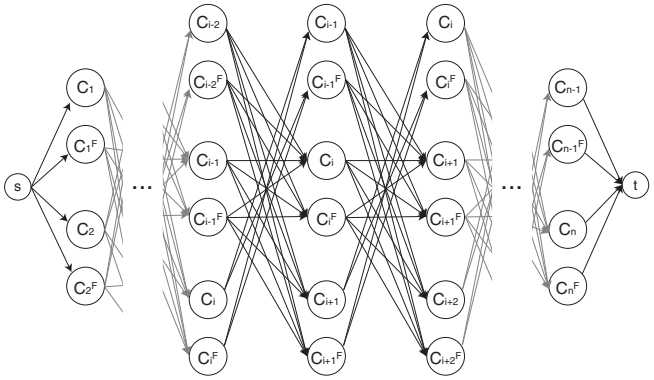

Fig. 11. Graph modeling to determine optimal cell swapping and flipping.

Our optimization is applied to placement rows one by one and consists of two options: (1) flip some cells along y-axis; (2) swap some adjacent cells as well as flip some. The second method benefits more but at the cost of increased wirelength due to larger cell displacement.

A. Cell Flipping

Let n cells in a row be denoted by $C_1, C_2, ..., C_n$ as illustrated in Fig. 10(a). We want to determine the orientation of each cell in a way that the number of whitespaces inserted in between cell pairs, whose DSA defect probabilities exceed the threshold, is minimized.

The problem can be modeled as the shortest path problem [14]. Each cell corresponds to two vertices, C_i and C_i^F, as shown in Fig. 10(b), where superscript F indicates that the cell is flipped. Two adjacent cells i and $i+1$ can be placed in 4 different configurations, which are represented by 4 edges. If a configuration requires a whitespace, i.e. the DSA defect probability of a cell pair exceeds the threshold, corresponding edge is associated with weight of 1; otherwise edge has 0 weight. Two dummy vertices, s and t, are added to finalize the graph.

The shortest path from s to t yields the solution. The graph we consider is directed acyclic graph (DAG) with nonnegative weights, so the complexity is $O(|V|+|E|)$ or $O(n)$ since $|V| = 2(n+1)$ and $|E| = 4n$. Let the shortest path from s to C_i be denoted by a list of vertices path(C_i). At C_{i+1}, we compare the path length of path$(C_i)\bigcup\{C_{i+1}\}$ and path$(C_i^F)\bigcup\{C_{i+1}\}$, and pick the path of shorter length for path(C_{i+1}); C_{i+1}^F can be obtained similarly.

B. Cell Swapping and Flipping

In addition to cell flipping, let us now assume that two adjacent cells can switch their positions. The problem can also be cast to the shortest path of DAG. As shown in Fig. 11, C_1 or C_2 can lead the placement row so 4 vertices are fanouts of s; similarly, C_{n-1} or C_n can be located at the end of row and 4

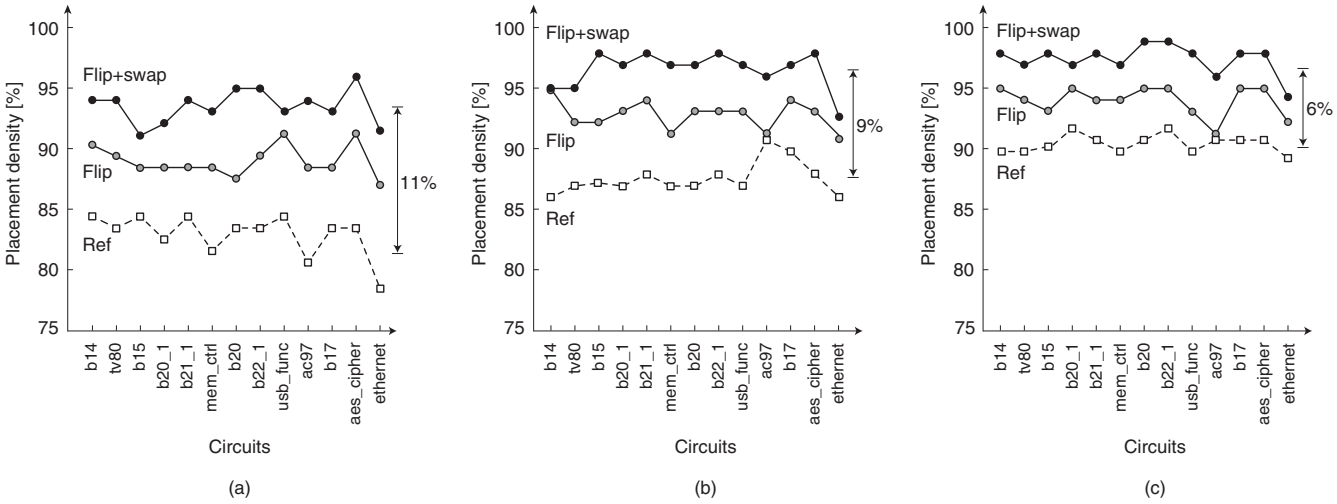

Fig. 12. Placement densities before (ref) and after (flip and flip+swap) our methods are applied, when the threshold of defect probability is (a) 0%, (b) 5%, and (c) 10%.

vertices are fanins of t. Other position of row corresponds to 6 vertices since C_{i-1}, C_i, or C_{i+1} can be located at i-th position. The graph is then finalized by inserting relevant edges, e.g. C_i in $(i-1)$-th position is connected to C_{i-1} and C_{i-1}^F in i-th position (i.e. C_{i-1} and C_i are swapped) but not to C_i and C_i^F (and C_{i+1} and C_{i+1}^F) in i-th position.

We may generalize cell swapping so that cells can be relocated further in a row. However, the problem is not formulated as the shortest path of DAG anymore. Imagine C_1, C_2, C_3, C_4, and C_5 and assume that each cell can be relocated to one or two positions before and after its original position. If C_3 is relocated to first position, it may or may not have an edge to C_1 (or C_1^F): if C_2 moves to third or fourth position, C_1 can be located at second so the edge is allowed; if C_2 does not move, C_1 can only be located at third position so the edge will not be drawn. Furthermore, cell swapping incurs the increase of wirelength as we will see in Section III-C, so this generalized cell swapping will not be tried.

C. Experiments

A few test circuits were extracted from Open Cores [15] and ITC99 [16] for experiments; the number of cells after logic synthesis ranges from 2k (b14) to 45k (ethernet). The proposed post-placement optimizations were implemented in Tcl script, which runs on commercial physical synthesis tool [17].

1) Assessment of Post-Placement Optimization: The two methods we propose are assessed in terms of placement density. We construct a reference placement for comparison. After the standard placement of each circuit, we scan placement rows; we insert whitespace in-between cell pair whose defect probability exceeds the threshold; if some rows have less whitespaces than necessary, placement is repeated with smaller density.

Placement densities are compared in Fig. 12(a) when the threshold is 0%. Cell flipping alone achieves 6% increase of density (from 82% to 88%) on average. Wirelength increases very marginally by 2%, so this method perturbs an initial placement very little. Cell swapping and flipping allows 11% increase of density, but at the cost of 9% increase of wirelength. Note that the wirelength is compared after actual routing.

The same comparison is also made when the threshold is 5% (Fig. 12(b)) and 10% (Fig. 12(c)). Placement density generally increases with increasing threshold because we need whitespaces in-between less number of cell pairs. Since our optimizations are applied to less number of cell pairs, the benefit clearly decreases.

2) Aspect Ratio of a Chip: Experiments in Fig. 12 were performed assuming square placement region. We repeat the experiments of Fig. 12(a) for two example circuits, but with varying aspect ratio of region. Fig. 13 shows the results. In the reference placement, the density decreases as the region becomes taller. This is because the width of region is determined by the widest row populated by cells and whitespaces (inserted between cell pairs of nonzero defect probability) and there are many rows (because the region is taller) having more whitespaces than necessary.

Initial placement is less perturbed by our methods as the region becomes taller, simply because each row contains less number of cells and so the candidates of optimizations also become sparse.

3) Library Pruning Using Defect Probability: A proactive approach to take DSA defect into account would be to consider it at logic synthesis. For each cell i in a library, we examine all cell pairs that include i; if 70% or more cell pairs have nonzero defect probability, i is dropped from library. About

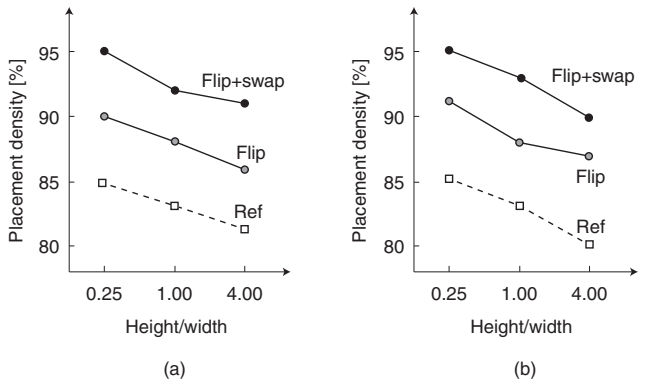

Fig. 13. Placement densities with varying aspect ratio of placement region: (a) b20_1 and (b) mem_ctrl.

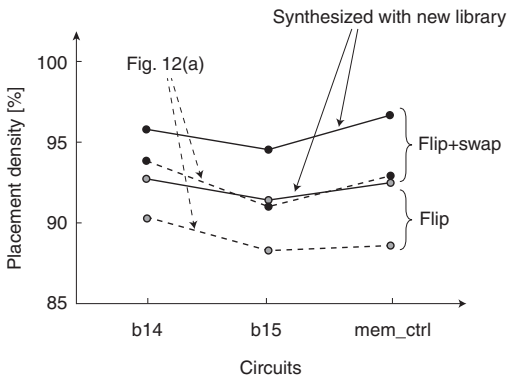

Fig. 14. Placement density (solid lines) when a circuit is synthesized with new library and is processed by our post-placement optimizations.

25% of cells (e.g. 1X_OR3 and 2X_AOI22) are removed in this process. All test circuits in Fig. 12 are re-synthesized using this new library; netlists are placed and are processed by our post-placement optimizations. The resulting placement density is compared to that from Fig. 12(a), and three sample circuits are shown in Fig. 14. An average of 4% increase of density is observed in both optimization methods.

IV. Conclusion

We have argued that inter-cell cluster of contacts should receive an attention in DSAL based design. This is because corresponding GP is not verified while each cell is designed and contacts are more likely to have defects in DSAL since they are patterned after two independent processes, lithography and DSA. All forms of inter-cell clusters can be identified in advance; they have been used to define the DSA defect probability of a cell pair in a library. We have introduced post-placement optimization, cell swapping and flipping, to respect a threshold on DSA defect probability while placement density is maximized. About 11% increase of density has been observed.

Instead of post-placement optimization we have tried, DSA defect may be taken care of while placement is being performed, which could be a topic for further study.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2015R1A2A2A01008037)

References

[1] H. Yi, Y. Bao, J. Zhang *et al.*, "Flexible control of block copolymer directed self-assembly using small, topographical templates: potential lithography solution for integrated circuit contact hole patterning," *Advanced Materials*, vol. 14, no. 23, pp. 3107–3114, Jul. 2012.

[2] M. Muramatsu, M. Iwashita, T. Kitano *et al.*, "Nanopatterning of diblock copolymer directed self-assembly lithography with wet development," *Journal of Micro/Nanolithography, MEMS, and MOEMS*, vol. 11, no. 3, pp. 1–6, Nov. 2012.

[3] W. Wang, L. A. Y. Zou, and T. H. Coskun, "A full-chip DSA correction framwork," in *Proc. SPIE Advanced Lithography*, Mar. 2014, pp. 1–11.

[4] L. Azat, G. Grant, P. Moshe, S. Gerard, W. Wong, J. Xu, and Y. Zou, "Computational simulations and parametric studies for directed self-assembly process development and solution of the inverse directed self-assembly problem," *Japanese Journal of Applied Physics*, vol. 53, no. 6S, pp. 1–8, May 2014.

[5] Y. Du, D. Guo, M. Wong *et al.*, "Block copolymer directed self-assembly (DSA) aware contact layer optimization for 10 nm 1D standard cell library," in *Proc. Int. Conf. on Computer Aided Design*, Nov. 2013, pp. 186–193.

[6] L. Liebmann, S. Mansfield, G. Han, J. Culp, J. Hibbeler, and R. Tsai, "Reducing DfM to practice: the lithography manufacturability assessor," in *Proc. SPIE Advanced Lithography*, Feb. 2006, pp. 786–798.

[7] J. P. Cain, "Design for manufacturability: a fabless perspective," in *Proc. SPIE Advanced Lithography*, Mar. 2013, pp. 1–9.

[8] H. Yi, A. Latypov, and P. Wong, "Computational simulation of block copolymer directed self-assembly in small topographical guiding templates," in *Proc. SPIE Advanced Lithography*, Mar. 2013, pp. 1–7.

[9] M. C. Smayling and V. Axelrad, "32nm and below logic patterning using optimized illumination and double patterning," in *Proc. SPIE Advanced Lithography*, Mar. 2009, pp. 1–10.

[10] "Nangate 15nm open cell library," http://www.nangate.com/.

[11] H. Yi, X. Bao, R. Tiberio, and P. Wong, "Design strategy of small topographical guiding templates for sub-15nm integrated circuits contact hole patterns using block copolymer directed self assembly," in *Proc. SPIE Advanced Lithography*, Mar. 2013, pp. 1–9.

[12] H. D. Ceniceros and G. H. Fredrickson, "Numerical solution of polymer self-consistent field theory," *Multiscale Model. Simul.*, vol. 2, no. 3, pp. 452–474, Feb. 2004.

[13] N. Laachi, K. T. Delaney, B. Kim *et al.*, "Self-consistent field theory investigation of directed self-assembly in cylindrical confinement," *Journal of Polymer Science: Part B Polymer Physics*, vol. 53, no. 2, pp. 142–153, Jan. 2015.

[14] Y. Du and M. D. F. Wong, "Optimization of standard cell based detailed placement for 16 nm FinFET process," in *Proc. Design, Automation and Test in Europe Conference and Exhibition*, Mar. 2014, pp. 1–6.

[15] "Opencores," http://www.opencores.org/.

[16] "ITC99," http://www.cerc.utexas.edu/itc99-benchmarks/.

[17] *IC Compiler User Guide*, Synopsys, Sep. 2014.